IN THE SPECIFICATION:

On page 1, prior to line 6, please insert the following headings and paragraph:

--Cross Reference to Related Applications

This application is for entry into the U.S. national phase under §371 for International Application No. PCT/IB03/01775 having an international filing date of May 7, 2003 and from which priority is claimed under all applicable sections of Title 35 of the United States Code including, but not limited to, Sections 120, 363 and 365(c), and which in turn claims priority under 35 USC §119 to Finnish Patent Application No. FI021759 filed on October 3, 2002.

Technical Field--

On page 1, prior to line 12, please insert the following heading:

--Background of the Invention--

On page 2, prior to line 8, please insert the following heading:

--Summary of the Invention—

On page 2, please amend the paragraph beginning at line 34 as follows:

--While other input methods (such as the T9 logic) make use of dictionary information, they do not require probabilities since characters are not selected completely one after another. Once a complete word is typed, ambiguities can be sorted out by presenting options to the user (in NOKIA phones by pressing the "*"-key). New words can simply be added to the list, which remains typically short even if the user has added many words to the dictionary.--

On page 8, please amend the paragraph beginning at line 16 as follows:

--According to the invention an indication (e.g. a usage parameter) reflecting the probability of occurrence of words (P_O(word)) in a text or in an input is added to each and all words in a dictionary. Secondly the number of words a user has input are counted as well as the number of occurrences for each word a user has input. It may be noted that the user most likely not

types in the words, as the availability of a complete keyboard dispenses the need for any kind of input prediction. The present invention may be especially useful in case of languages with an alphabet that is based on syllables, like Japanese Hiragana and Katakana. The more single elements can be found in the alphabet of a language the more useful the present invention of prediction can be applied to simplify the input of a character. Especially in Japanese the number of [[turn]] turns of a thumb wheel should be reduced to enable a useful and applicable thumb-wheel input algorithm. The present invention can simplify the input of words and text. Based on the total number of words and the individual number of occurrences the probability of occurrence is estimated. This may involve additional heuristics rather than pure arithmetic in particular if the total number of words is still small to adapt quickly to user behavior although the statistical variance of the probability estimate is still very big.--

On page 9, please amend the paragraph beginning at line 19 as follows:

--In addition to words, also diphones (or syllables as in the Japanese language) can be used if no words are known that extend the characters typed so far. It is necessary to assign as well probabilities to those to avoid that they are completely removed if the user dictionary grows, although the probabilities of the involved words are very low.--

On page 11, please amend the paragraph beginning at line 11 as follows:

--In ~~an other~~ another example embodiment, said input prediction with said determined inference logic comprises identifying a start of an entry of a new word, inserting, into the character subset, the most probable letters stored in the database of words for starting a word. This can be embodied by selecting the nodes with the highest probability in the first level.--

On page 11, please amend the paragraph beginning at line 16 as follows:

--The present invention can be extended by suppressing single words in the database for the second or a few following entries, as it can be expected that a user is not going to use the same word for several times in direct succession. In case the device provides enough storage and processing power the present invention can also be extended with probabilities for the successions of different words. So it can easily be prevented that same or like words follow

each other, and additionally frequently used idiomatic expressions or phrase can be recognized with an increased probability. An easy way to provide probabilities for successive words is to provide a probability tree with a two-word depth. That is, two successive words are regarded as a single word and the probability for the succession of the two words is coded in the nodes of a successive tree. As stated above, the definition of a word as a string of characters also comprises the definition of two successive strings of characters and (e.g. a "space") in between as a word. So in the case that a second word in a sequence is usually comprised of a specific subset of word as e.g. in starting a message "Hi, Suzie...", that is two starting words can be regarded as a single sequence of characters (i.e. a word). So the Device is able to predict the next input by proposing one of "Hi, Bernd...", "Hi, Jack..." and the like. Depending on the number of usage parameters, and the processing power provided in the device, the device may also be able to predict that a word following an expression such as "Hi, ...", "Hello, ..", "AVE ...", "Salu ..." or the like is most probably to be followed by a name. It is clear that such a sophisticated provision [[of]] requires [[many]] a great amount of statistic information about a language. [[Incas]] In case of a telephone, this would only require a small short cut to the telephone register, although this may lead to message heading [[s]] such as "My sweet, Jody mobile ... ".—

On page 12, please amend the paragraph beginning at line 2 as follows:

--Another example embodiment of said method is characterized in that said determined inference logic comprises identifying a text to be entered, identifying words being stored in said database of words that are appropriate for the word or text being entered and selecting, for the character subset, a character from each appropriate probable entry to be possibly entered next. That is the search for the most probable character(s) continues to the next level of nodes in the probability tree.--

On page 12, please amend the paragraph beginning at line 36 as follows:

--The software prediction module can be implemented to run on a controller and to use a ~~non volatile~~ non-volatile memory to store the frequency of number dialing and a list of all dialed numbers.--

On page 15, prior to line 3, please insert the following heading:

--Brief Description of the Drawings--

On page 15 prior to line 35, please insert the following heading:

--Detailed Description--

On page 24, please amend the paragraph beginning at line 21 as follows:

--It is also possible that the system only ~~suggest only~~ suggests the most probable next character of digit as the most probable next user input. That is the expression "browsing" in the claims describes the selection or non-selection of a single proposed character. This can be very useful if the user is not used to use the proposed selection, offering an easy way out to standard character selection. Although this may provide only a single option, [[as]] a single ~~proposes~~ proposed character is sufficient for the present invention. The advantage of the present invention resides in a faster input of digits in the case the prediction comes up with the [[right]] correct digits, and there is no disadvantage if the prediction suggests ~~a wrong~~ an incorrect digit, ~~expect of~~ except for the fact that a user has to scroll away from said suggested subset.--

On page 24, please amend the paragraph beginning at line 31 as follows:

--Other advantages reside in that

- the dictionary is learning enabled (or at least made very effective) together with the predictive input,
- the user experience is improved,
- the effectiveness of predictive input is increased
- the methods that try to implement a similar user experience without involving probabilities will have to be based on heuristics that will most likely lead to strange side effects (since the system cannot really know[[,]] how often a user will use a certain word without counting) such as proposing rare words all the time and not recognizing/proposing frequent words.
- that all new words can be added automatically to the user dictionary since words that occur only once in a lifetime will not spoil the prediction engine due to their low probability.

- the method is based on probabilities and a user dictionary of limited size that is easier to maintain since rare words are known and can be removed. However, new words need to remain for some time to permit that the probability can build up before the word is removed due to its low probability.--